BUNDESREPUBLIK DEUTSCHLAND

Bescheinigung

Die Philips Patentverwaltung GmbH in Hamburg/Deutschland hat eine Patentanmeldung unter der Bezeichnung

"Smart-Card-Controller mit optimaler Anpassung an die zur Verfügung stehende Versorgungsenergie"

am 30. Spetember 1998 beim Deutschen Patent- und Markenamt eingereicht.

Der Firmenname der Anmelderin wurde geändert in:
Philips Corporate Intellectual Property GmbH.

Das angeheftete Stück ist eine richtige und genaue Wiedergabe der ursprünglichen Unterlage dieser Patentanmeldung.

Die Anmeldung hat im Deutschen Patent- und Markenamt vorläufig die Symbole G 06 K und G 06 F der Internationalen Patentklassifikation erhalten.

München, den 25. November 1999
Deutsches Patent- und Markenamt
Der Präsident
Im Auftrag

Aktenzeichen: 198 45 022.2

Dzierzon

PRIORITY DOCUMENT
SUBMITTED OR TRANSMITTED IN
COMPLIANCE WITH
RULE 17.1(a) OR (b)

A 9161
06.90
11/98

Smart-Card-Controller mit optimaler Anpassung an die zur Verfügung stehende Versorgungsenergie

Anwendungsgebiet, sowie Beschreibung des Stands der Technik

Smart-Card-Controller mit mindestens einem kontaktlosen Interface, aber auch solche mit mehrfacher Interfacestruktur erobern einen bedeutenden Teil des Marktes für intelligente Karten. Diese haben z.B. sowohl ein kontaktloses Interface (z.B. entsprechend ISO 14443), als auch (optional) ein kontaktbehaftetes Interface (z.B. entsprechend ISO 7816).

Je logisch mächtiger die unterlegte Controller-Struktur ist, desto schwieriger wird dabei die Lösung der Energieversorgungsprobleme, da die mächtigere Struktur einen höheren Versorgungsstrom ziehen wird. Durch das hier angenommene kontaktlose Interface bedingt, muß die Schaltung mit sehr geringen Versorgungsströmen auskommen, die in der Größenordnung von 1 mA liegen. Mit der daraus resultierenden Versorgungsleistung müssen alle Schaltungsteile des Controllers gespeist werden können.

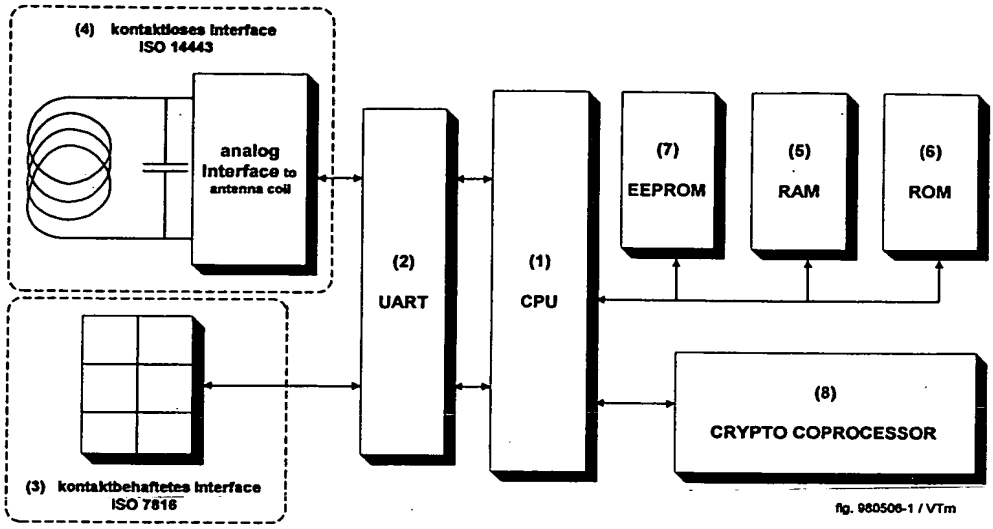

Bild 1: typischer Smart-Card-Controller mit doppeltem Interface

Bild 1 zeigt einen typischen Smart-Card-Controller mit doppeltem Interface. Er besteht üblicherweise aus den Modulen

(1) CPU (Verarbeitungseinheit),

(2) UART (universeller asynchroner Receiver-Transmitter),

(3) (optional) kontaktbehaftetes Interface zur Außenwelt,

(4) kontaktloses Interface (Kopplung über elektromagnetisches Feld),

(5) Variablenspeicher RAM (Random Access Memory),

(6) Programmspeicher ROM (Read-Only Memory),

(7) nichtflüchtiger Speicher EEPROM (electrical erasable programable Read-Only Memory),

(8) (optional) Crypto Coprozessor.

Die Leistungsversorgung der Gesamtschaltung geschieht jeweils über das aktuell im Betrieb befindliche Interface, also entweder über das kontaktbehaftete Interface oder alternativ über das kontaktlose Interface.

Fall (1) Kontakt-Interface (ISO 7816)

Hier besteht eine Limitierung durch die Spezifikation des ISO7816. Der Strom in den Versorgungsanschluß ist per ISO-Standard auf 50 mA begrenzt (ISO 7816-3 @ 5 MHz).

Für moderne integrierte Schaltungen in CMOS-Technik, die in Hinsicht auf geringen Leistungsverbrauch (Low Power) ausgelegt sind, stellt dies im allgemeinen keine schwerwiegende Begrenzung dar.

Fall (2) Kontaktloses Interface (z.B. ISO 14443)

In diesem Fall geschieht die Speisung über das elektromagnetische Feld. Bei Anwendung des kontaktlosen Interfaces von ISO 14443, Typ A läßt sich bei (heute üblichen Operationsdistanzen von etwa 10 cm) so ein Strom von minimal 1 mA (maximal ca. 2 mA) übertragen. Dieser geringe Versorgungsstrom muß zur Speisung aller Module in allen Aktivitäten ausreichen.

Aufgabe der Erfindung

Die Aufgabe der Erfindung besteht nun darin, für Anwendungen im Smart-Card-Bereich aber auch andere leistungssensitive Bereiche eine Realisierung für die Verarbeitungseinheit zu finden, die mindestens folgende Eigenschaften enthält:

- Der Controller soll mit optimal zwei Interfaces (zum Beispiel kontaktbehaftet und kontaktlos) arbeiten.
- Die Performance des Controllers inklusive der Speicher und eventuell des Coprozessors soll optimal den energetischen Gegebenheiten angepaßt werden, d.h.
  - (a) der Controller kann nicht mehr Versorgungsenergie verbrauchen als bereitgestellt wird,
  - (b) der Controller soll die bereitgestellte Energie optimal ausnutzen.
- Im Optimaleinsatz soll eine gegebene Funktionseinheit so

schnell arbeiten, wie es für ihren Einsatz nötig ist und dabei nur den Anteil an Energie konsumieren, der dafür theoretisch minimal notwendig ist.

Als Beispiel soll hier der parallele Betrieb des Controllers und der Crypto-Einheit dargestellt werden.

In Bild 2 ist dargestellt, wie die Verarbeitungseinheit CPU (1) und der Crypto-Coprozessor (8) mit den Speichern (5), (6), (7) verbunden sind.

Während des Vorgangs einer Berechnung zur Verschlüsselung/ Entschlüsselung ist der Coprozessor prinzipiell für die Größenordnung von einigen Millisekunden durchgehend beschäftigt, während die CPU in dieser Zeit nur wenige Pointer neu laden muß. Sie hat also in der Zeit nahezu nichts zu tun. Man kann nun ihre Versorgungsspannung für die CPU so weit herunterfahren, daß sie diese (wenige) Arbeit in einer längeren Zeit (als sonst) abwickelt und dabei auch erheblich weniger Verlustleistung produziert.

Die CPU arbeitet dann in einem Low-Performance-Mode und erledigt trotzdem ihre Arbeit rechtzeitig.

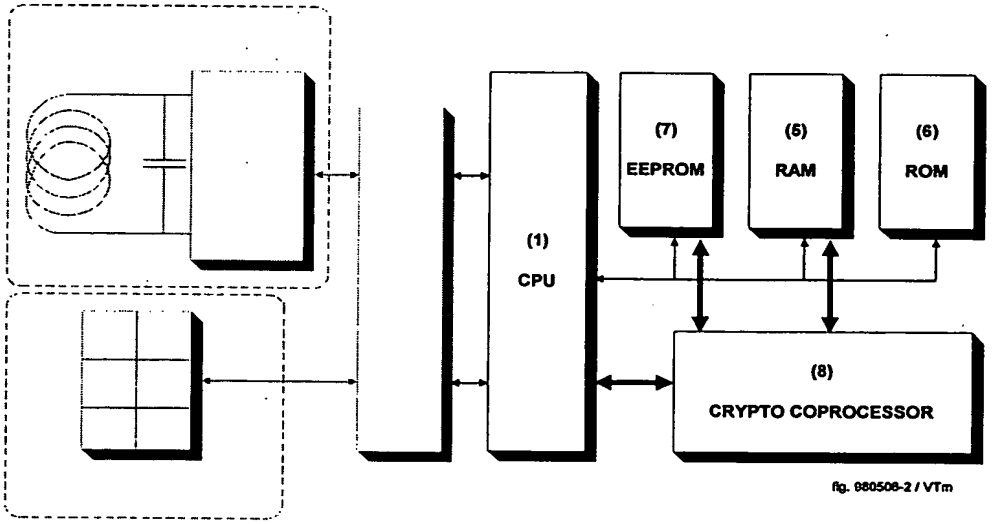

Bild 2: Betrieb von CPU und Crypto-Controller

Weitere Forderungen

- Der Energieverbrauch soll derart gesteuert werden, daß mehrere Funktionseinheiten mit der aktuell zur Verfügung stehenden Energie auskommen und dabei eine für die bereitstehende Energie optimale Performance erreichen.
- Die Module sollen in mehreren Verarbeitungsmodi arbeiten können, die sich jeweils durch die verbrauchte Energie und die Verarbeitungsgeschwindigkeit unterscheiden.
- Eine Zufälligkeit der Länge der Ausführungszeit soll hergestellt werden.
- Beim Betrieb soll eine automatische Synchronisation zur Außenwelt sichergestellt werden.
- Alternativ arbeiten die Funktionseinheiten nacheinander. Damit kann vermieden werden, daß durch Überlagerung der Stromverbräuche aller Einheiten zuviel Strom verbraucht wird. Beim Betrieb der Einheiten nacheinander werden die Einheiten derart ausgelegt, daß ihr individueller Stromverbrauch nicht größer ist, als der von der Versorgung minimal bereitgestellte Wert.

Erfindungsgemäße Lösung der gestellten Aufgabe

Erfindungsgemäß wird die Aufgabe durch folgende spezielle Struktur gelöst:

- Die Verarbeitungeinheit wird in asynchroner (delay-orientierter) Logik realisiert [1], [2].

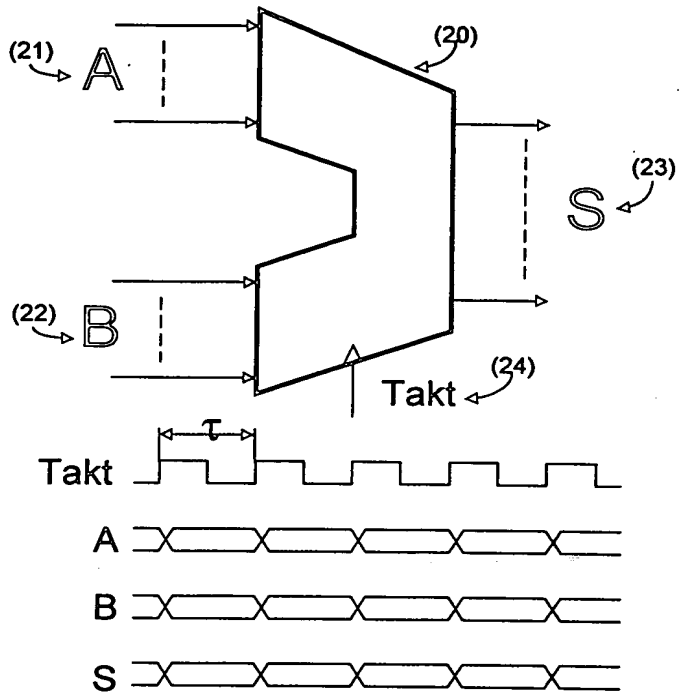

Bild 3: Beispiel für ein synchrones Funktionsmodul

Zur Erklärung der asynchronen Funktionsweise ist in Bild 3 zunächst ein synchron betriebenes Modul angegeben. Es arbeitet wie folgt:

Das Funktionsmodul (20) wird von zwei Operanden (21) und (22) beschickt. Zum Zeitpunkt der aktiven Flanke des Taktes (24) werden beide Operanden im Modul verknüpft und auf den Ausgang (23) durchgeschaltet.

Diese Betriebsweise wird als synchron bezeichnet, weil sie synchron mit dem Takt erfolgt. Das Zeitverhalten des Taktes bestimmt die Betriebsweise des Moduls.

Bild 4 zeigt im Gegensatz dazu eine asynchrone Betriebsweise.

Das asynchrone Funktionsmodul (30) wird durch die Operanden (31) und (32) beschickt. Auf Anforderung der Leitung REQ (request) (35) wird die Operation des Moduls aktiviert. Wenn das Modul mit seiner Operation fertig ist

meldet es sich eigenständig auf der Leitung ACK (acknowledge) (34) mit einer Fertigmeldung. Zu diesem Zeitpunkt wird das Ergebnis der Operation auf die Leitungen (33) durchgeschaltet.

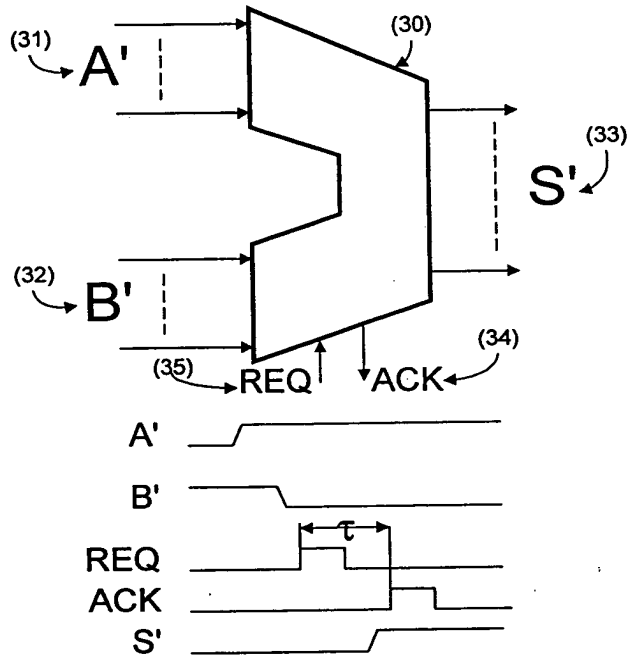

Bild 4: Asynchrones Funktionsmodul

- Asynchrone Logik hat folgendes Verhalten:
  - Die Logik arbeitet ohne Takt, allein delay-orientiert;
  - Die Logik arbeitet handshake-gesteuert auf Anforderung (request) und antwortet nach durchgeführter Aktivität mit einer Fertigmeldung (acknowledge). Dabei wird die Gesamtlogik einer Schaltung eventuell in kleinere Module zerlegt, deren asynchrone Realisierung überschaubar ist. Die Module kommunizieren untereinander im Handshake-Verfahren;
  - Mit abnehmender Versorgungsspannung nimmt der Delay der Logik nichtlinear zu (s. Bild 5). Die Abhängigkeit ist bei CMOS-Logik üblicherweise

quadratisch.

- Die Speicherblöcke sind dabei selbst-steuernd (self-timed). Die Speicher arbeiten in ‚hand-shake'-Betriebsweise.
- Die Verarbeitungseinheit sowie alle nötigen Module erhalten mehrere Verarbeitungsmodi, die sich in ihrem Energieverbrauch und daraus folgernd in ihrer Verarbeitungsgeschwindigkeit unterscheiden.
  Dabei sind mehrere Realisierungen möglich:

  (a) Gestufte Einstellung des Energieverbrauchs über eine gestufte Einstellung der Versorgungsspannung;

  (b) Stufenlose Anpassung des Energieverbrauchs durch eine stufenlose Einstellung der Versorgungsspannung, z.B. mit Rückkopplung als Regler.

  Durch die Eigenheit der asynchron arbeitenden Logik, zu jeder Höhe der Versorgungsspannung eine dazu passende Verarbeitungsgeschwindigkeit zu erreichen, kann die gewünschte Verarbeitungsleistung durch Einstellung der Versorgungsspannung erreicht werden.

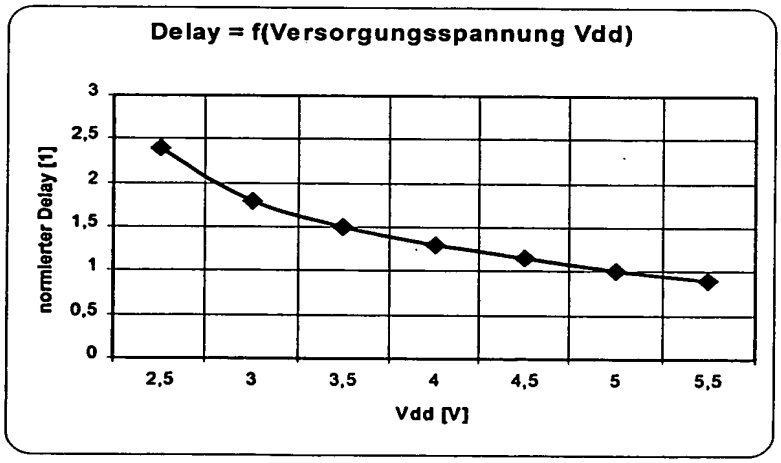

Bild 5: normierter Delay einer CMOS-Gatterschaltung als Funktion der Versorgungsspannung Vdd

- Der Betrieb der Verarbeitungseinheit kann sowohl

  (a) Völlig asynchron als auch

  (b) Quasi-synchron, gesteuert durch einen bereitzustellenden Synchronisations-Takt, erfolgen. Hierzu wird an Ereignisgrenzen ('Event'-Grenzen), die aus einem Zeitgeber (Timer) bereitgestellt werden und (in etwa) den sonst synchron erzeugten Instruktionsgrenzen entsprechen, eine Synchronisation des Instruktionsablaufs hergestellt, so daß sich der Controller außerhalb der Instruktionsgrenzen so 'benimmt', als wäre er synchron betrieben.

  Die Betriebsweise (b) ist beim Fehlersuchen (Debuggen) der Software behilflich. Sie kann danach mit einem geeigneten Schalter abgestellt werden.

- Die völlig asynchrone Betriebsweise erzeugt Ausführungszeiten, die in ihrer Länge nicht vorhersehbar sind, so daß feindliche Angriffe auf das Chip, die sich auf das Ermitteln der Ausführungszeiten ausrichten, hierdurch unmöglich gemacht werden.

- Die völlig asynchrone Betriebsweise macht weiterhin eine Attacke über eine Differential Power Analysis (DPA) unmöglich. Diese Methode macht Gebrauch von Mustern, die durch die Betriebsweise der Schaltung auf der Versorgungsleitung abgebildet werden. Durch Korrelation verschiedener Muster, die bei verschiedenen Datenmustern erzeugt werden, sollen mit dieser Methode Rückschlüsse auf die verarbeiteten Daten erreicht werden. Da dies den streng synchronen (taktgesteuerten) Ablauf von Operationen voraussetzt, kann eine asynchrone, nicht an einem Takt synchronisierte Betriebsweise die Methode DPA unmöglich machen.

- Die Synchronisation der Verarbeitungseinheit zur Außenwelt geschieht über einen integrierten UART (universeller asynchroner Receiver/Transmitter). Dieser

kann sowohl per Hardware als auch per Softwaremodul erstellt werden. Die Realisierungsform richtet sich allein nach der geforderten Durchsatzleistung (Geschwindigkeit der seriellen Schnittstelle) relativ zur Verarbeitungsleistung der Verarbeitungseinheit.

Ein Übergangspunkt zwischen der software-basierten und der hardware-basierten Lösung muß (nach heutigen Maßstäben) etwa zwischen 50 bis 100 k Baud seriellem Durchsatz angesiedelt werden, d.h. bei einer geforderten Durchsatzleistung $\leq$ 50 k Baud genügt im allgemeinen ein software-gestützter UART, während darüber ein hardware-gestützter UART einzusetzen ist.

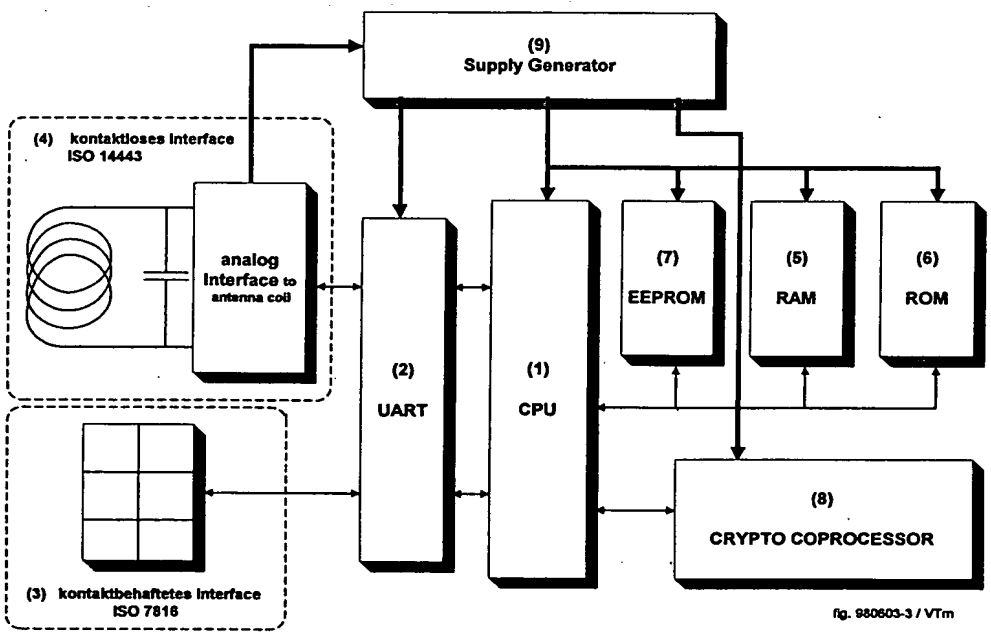

Bild 6: Versorgungsgenerator

Bild 6 zeigt einen Generator, der die Versorgungsspannungen zum Betrieb der Module nach obigen Verfahren generiert.

Bild 7 zeigt als Speisegenerator eine Stromquelle (10), welche die Last aus parasitärem Kondensator (11) und Widerstand (12) speist. Dabei repräsentiert der Widerstand die Logik, die allerdings nur sporadisch Strom dann zieht, wenn sie aktiv ist. Der Widerstand ist also variabel.

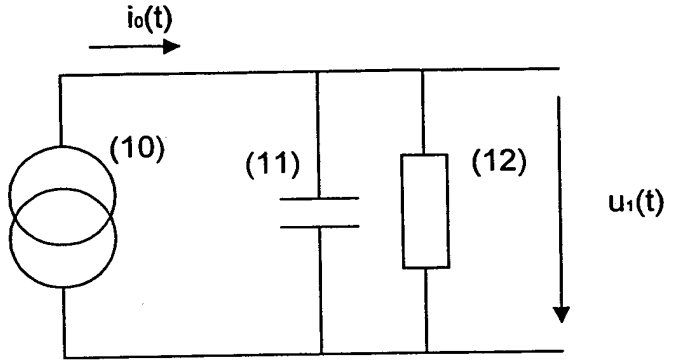

Bild 7: Ersatzschaltbild der Versorgungsschaltung (ausgebildet als Stromquelle)

- Die Stromquelle, die durch das elektromagnetische Feld gespeist wird, speist ihrerseits die Last aus Widerstand und Kondensator. Sowohl Stromquelle als auch ohm'sche Last sind Variablen über der Zeit. Wenn jetzt die Aktivität der Logik steigt, weil sie „mehr zu tun bekommt", so sinkt die resultierende Versorgungsspannung über der Last, da der Strom in einen kleineren (fiktiven) Widerstand fließt. Mit verringerter Versorgungsspannung nimmt der Delay der Logik zu (s. Bild 5). Die Logik wird dadurch langsamer, damit sinkt ihre Aktivität, damit steigt die Versorgungsspannung wieder an, da der (fiktive) Widerstand steigt. Dieser Kreis ist selbstregelnd.

Patentansprüche

Hauptanspruch 1. Datenverarbeitungseinheit mit asynchroner (delay-orientierter) Verarbeitungsweise für Chipkarten mit kontaktbehaftetem und/oder kontaktlosem Interface derart ausgerüstet, daß eine gegebene Ausführungszeit ausgenutzt wird und nur die dafür nötige Versorgungsleistung verbraucht wird.

2. Datenverarbeitungseinheit mit asynchroner (delay-orientierter) Verarbeitungsweise, welche die Performance der Verarbeitungseinheit optimal den gegebenen Energieverhältnissen anpaßt.
3. sowohl asynchroner als auch quasisynchroner Betrieb
4. Synchronisation zur Außenwelt mittels UART und mittels von außen getakteten Zeitgebermitteln.
5. Durch die asynchrone Logik hervorgerufene Zufälligkeit der Ausführungszeit.
6. Zeitlich gestaffeltes Anschalten der Verarbeitungseinheiten (Betrieb nacheinander) im Gegensatz zu gleichzeitiger Betriebsweise.
7. Benutzung einer Versorgung in Form einer Stromquelle. Automatische Adaption der Versorgungsspannung.
8. Verhinderung von Differential Power Analysis bei der Ausforschung der Verarbeitungsabläufe der Schaltung.

Literatur

[1] Kees van Berkel: Handshake Circuits, An asynchronous architecture for VLSI programming, ©Cambridge University Press 1993,
ISBN 0 521 45254 6

[2] Ad. M. G. Peeters, Single Rail Handshake Circuits, Proefschrift Technische Universiteit Eindhoven, ISBN 90-74445-28-4